

# SEG3102 – Fall 19

## Project Deliverable #3

### Design/Implementation

In this deliverable, you will complete the design of the TMS system and provide an implementation. Because of the size of the project, only the following use cases are required in the implementation: *Register User, Instructor log in, Set up parameters, Visualize students teams, Instructor log out, Student log in, Create team, Join team, Accept new students, Student log out.*

You are asked to submit a design document and the code corresponding to the implementation of your design.

***Note: it is important that your implementation code correspond to your design. You will have to show traceability in your deliverable document.***

In addition to the ***standard elements***, you will submit a document with the following outline:

1. Executive Summary: Briefly state what the document is about and what are its contents.
2. Description of team and roles: Provide a table with all team members and the tasks they were involved in for this deliverable. You can be specific by stating percentages of the work done and number of hours spent by each team member if this information is available. If for some legitimate reasons, the work share is unbalanced, please state these reasons in a short paragraph. *Notice that the information provided here might be used to adjust team members final grades for the project.*
3. Assumptions and Design Decisions: List all assumptions as well as all design decisions made.
4. Analysis elements – these are mainly for completeness. Provide updated elements from your previous deliverable
  1. Analysis-level UML Class Diagram : Provide the UML Class Diagram from deliverable 2 (updated or not).
  2. Analysis Mechanisms from deliverable 1 (updated or not).
  3. User Experience Model - Screen definition and Navigation map from deliverable 1 (updated or not).
  4. Analysis level use case realizations from deliverable 2 (updated or not).
5. Design and Implementation mechanisms – show how your analysis mechanisms are realized in term of design and implementation mechanisms. Provide a brief rationale for each decision.
6. Design Elements
  1. Provide UML Class diagrams showing your presentation layer elements and their dependencies.
  2. Provide a UML diagram (a UML Component Diagram or a UML Package Diagram) showing each of your Subsystems and their dependencies.
  3. For each Subsystem, provide a UML diagram (a UML Component Diagram and/or UML Class Diagram) showing its elements (e.g. EJB, Entities, Classes, Interfaces) with associations and dependencies.
7. Design Level Use Case Realizations: Provide UML Sequence Diagrams showing the realization of each use case in the current iteration. These are mappings of the analysis-level Sequence Diagrams on the identified design elements.
8. Implementation - Traceability information between design elements and code – show how the design elements (from sections 5, 6) trace to the provided source code.

For instance, you may provide a table with the following format

Design Element	Implementation Element (Java EE) <sup>1</sup>
Implementation mechanism 1	Packages/Classes/Methods corresponding (or « none » if not implemented)
Presentation layer element 1	Package/Classe/ or JSF corresponding (or « none » if not implemented)
Subsystem 1	Package corresponding (or « none » if not implemented)
Framework Component (Entities, EJBs, Webservices, ...) 1	Package/Classe corresponding (or « none » if not implemented)

- Use the **@Table** annotation to specify a custom name to the tables corresponding to your entities – if Java EE implemented. For instance :

```
@Entity  
@Table(name="CUSTOMERTABLE1234567")  
public class Customer {...}
```

Where *1234567* is a team member student ID.

9. Discussion – provide a short discussion on how your design promotes the quality of *modifiability* (adding/altering use cases).
10. References.

<sup>1</sup> Provide the corresponding according to your implementation Framework, if you haven't used JavaEE.